The Untold Secrets of WiFi-Calling Services: Vulnerabilities, Attacks, and Countermeasures

Tian Xie, Guan-Hua Tu, Bangjie Yin, Chi-Yu Li, Chunyi Peng, Mi Zhang, Hui Liu, Xiaoming Liu

Abstract—Since 2016, all of four major U.S. operators have rolled out Wi-Fi calling services. They enable mobile users to place cellular calls over Wi-Fi networks based on the 3GPP IMS technology. Compared with conventional cellular voice solutions, the major difference lies in that their traffic traverses untrusted Wi-Fi networks and the Internet. This exposure to insecure networks can cause the Wi-Fi calling users to suffer from security threats. Its security mechanisms are similar to the VoLTE, because both of them are supported by the IMS. They include SIM-based security, 3GPP AKA, IPSec, etc. However, are they sufficient to secure Wi-Fi calling services? Unfortunately, our study yields a negative answer. We conduct the first security study on the operational Wi-Fi calling services in three major U.S. operators' networks using commodity devices. We disclose that current Wi-Fi calling security is not bullet-proof and uncover three vulnerabilities. By exploiting the vulnerabilities, we devise two proof-of-concept attacks: telephony harassment or denial of voice service and user privacy leakage; both of them can bypass the existing security defenses. We have confirmed their feasibility using real-world experiments, as well as assessed their potential damages and proposed a solution to address all identified vulnerabilities.

Index Terms—Wi-Fi calling, security and privacy, computer vision recognition, and cellular network.

1 Introduction

Since 2016, all the four major operators in the U.S., namely T-Mobile, AT&T, Verizon and Sprint, have launched nationwide Wi-Fi calling services [1]. The Wi-Fi calling technology, also known as VoWiFi (Voice over Wi-Fi), is supported by the 3GPP IMS (IP Multimedia Subsystem) system [2]. It provides mobile users with cellular calls and text messages through home/public Wi-Fi access networks instead of cellular base stations. It is an alternative voice solution for mobile users that connect to the base stations with weak signals. Globally, there had been 98 cellular network operators offering Wi-Fi calling services in 52 countries [3] until February 2019. According to a recent industry report [4], the trends that about 71% of mobile data will go through Wi-Fi networks and about 80% of mobile users will use Wi-Fi to access the Internet, will result in a rising demand for the Wi-Fi calling market. The market is forecasted to grow at 27.24% CAGR (Compound Annual Growth Rate) to over 8 billion U.S. dollars by 2025 from 1.92 billion in 2020. With such rapidly growing market, any security loopholes of Wi-Fi calling may lead to devastating consequences on a global scale. Therefore, there is a critical need to investigate the security of Wi-Fi calling.

Wi-Fi calling uses SIP (Session Initiation Protocol) for the call signaling as conventional VoIP (Voice over IP) services do, but differs from them technically. Its SIP signaling operation is a 3GPP-specific version [5], [6]. For security reasons, both 3GPP and GSMA stipulate that Wi-Fi calling shall use well-examined SIM-based security and authentication methods as VoLTE has. They mainly include the protection of secret keys in a physical SIM card and the 3GPP AKA (Authentication and Key Agreement) [7] authentication. In addition, all the Wi-Fi calling packets, which may be sent through insecure networks, shall be delivered via the IPSec (Internet Protocol Security) channels using ESP tunnel mode between Wi-Fi calling devices and the cellular network infrastructure. Although the packets are protected by the IPSec tunnels, the Wi-Fi calling service may still suffer from DoS (Denial-of-Service) attacks where the packets are maliciously dropped en route. However, such DoS attacks can be prevented by the inter-system switch security mechanism of Wi-Fi calling, which switches a Wi-Fi calling user back to the cellular-based voice/text service when the user is unreachable through Wi-Fi.

When adopting the conventional security mechanisms which have been well studied in VoLTE [10], [11], Wi-Fi calling seems to be as secure as VoLTE. Unfortunately, it is not the case. We discover three security vulnerabilities from all the Wi-Fi calling services deployed by three cellular network operators in the U.S. and two operators in Taiwan, which are denoted as US-I, US-II, US-III, TW-I, and TW-II, respectively. First, the 3GPP WLAN (Wireless Local Area Network) selection mechanisms, which are used to select a Wi-Fi network for the Wi-Fi calling device, do not prevent devices from connecting to insecure Wi-Fi networks (V1), which may impede the Wi-Fi calling service. Second, the Wi-Fi calling traffic, which is protected by IPSec, is vulnerable to side-channel inference attacks (V2), which may cause privacy leakage. Third, the service continuity mechanism between Wi-Fi calling and cellular-based voice services may

- *T. Xie, G.-H. Tu, B. Yin, H. Liu, and X. Liu are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI, 48825. E-mail:{xietian1, ghtu, yinbangj, liuhui7, liuxm}@msu.edu.*
- *M. Zhang is with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48825. E-mail:mizhang@msu.edu.*
- *C.-Y. Li is with the Department of Computer Science, National Chiao Tung University. E-mail:chiyuli@cs.nctu.edu.tw*
- *C. Peng is with the Department of Computer Science, Purdue University, West Lafayette, IN, 47907. E-mail:chunyi@purdue.edu*

Category	Vulnerability	Type	Root Cause
Device	V1: the WLAN selection mechanisms of Wi-Fi calling devices do not prevent the devices from connecting to insecure Wi-Fi networks.	Design defect	The 3GPP standard [1], [8] considers only the radio quality of available Wi-Fi networks without any security measures in the WLAN selection (Section 4.1).
Infrastructure	V2: the Wi-Fi calling traffic is vulnerable to side-channel inference attacks.	Operation slip	The IPSec sessions between Wi-Fi calling devices and the core network carry only the Wi-Fi calling traffic, so its traffic patterns can be learned to infer various Wi-Fi calling events (Section 4.2).
	V3: the service continuity mechanism between Wi-Fi calling and cellular-based voice services may not take effect when needed.	Design defect	The service continuity mechanism based on an inter-system switch [1], [2], [9], which keeps a call service continue across different radio access technologies, considers only radio quality but not service quality (Section 4.3).

TABLE 1
Summarizing the identified security vulnerabilities of the Wi-Fi calling services.

not take effect (V3), even when the service quality of a Wi-Fi calling call is so bad that its voice is almost muted. Each of these vulnerabilities can be attributed to a design defect of the Wi-Fi calling standard or an operational slip of the cellular network. Table 1 summarizes the vulnerabilities and their root causes.

We exploit the three vulnerabilities to devise two proof-of-concept attacks, namely (1) telephony harassment or denial of voice service attack (THDoS) and (2) user privacy leakage. These two attacks can bypass the existing security mechanisms on the Wi-Fi calling devices and the cellular network infrastructure. In the first attack, we devise four attack variants that harass Wi-Fi calling users or get them denial of voice services. In the second attack, we develop a user privacy inference system (UPIS) that incorporates the face recognition technique in computer vision with the exploitation of those vulnerabilities. The UPIS system can disclose the privacy of a Wi-Fi user, including user identity, call statistics, and the device's IP address. Particularly, the call statistics have been proven effective in inferring a user's personality [12] (e.g., conscientiousness), mood [13] (e.g., stressful), and behavior [14] (e.g., dialing spamming calls). With the inferences of the device's IP address and the user identity of a Wi-Fi calling user, adversaries can discover the user's device model, Internet activities (e.g., accessing CNN.com), and the device's running applications by analyzing his/her packets. Note that different from traditional SIP attacks [15]–[17], the proposed attacks not only need to identify particular Wi-Fi calling signaling messages from encrypted 3GPP-specific SIP packets [5], [6], but also have to bypass/suppress cellular-specific security mechanisms such as the inter-system switch mechanism that keeps the Wi-Fi calling service continuity.

We finally propose a solution, Wi-Fi Calling Guardian, to address these security threats, without requiring any modifications to Wi-Fi calling standards, which is unlikely to be achieved in a short time. In summary, this paper makes four key contributions.

- We conducted the first security study to explore the dark side of operational Wi-Fi calling services in five operational cellular networks in the U.S. and Taiwan using commodity devices. We identified three Wi-Fi calling vulnerabilities, each of which roots in a design defect of the Wi-Fi calling standard or an operational slip of the operators.
- We devised two proof-of-concept attacks by exploiting the identified vulnerabilities and assessed their negative impacts in a responsive manner.
- We developed a practical solution, Wi-Fi Calling Guardian, to address the identified vulnerabilities. Our experiments confirm that it can protect the Wi-Fi calling users from the proposed security threats.
- We actively reported and demonstrated the security threats to the industry, and received a positive feedback. Specifically, the security team of Google Android has confirmed our findings and promised to address the vulnerability that coming from the device. Our research result can thus benefit billions of Android phone users.

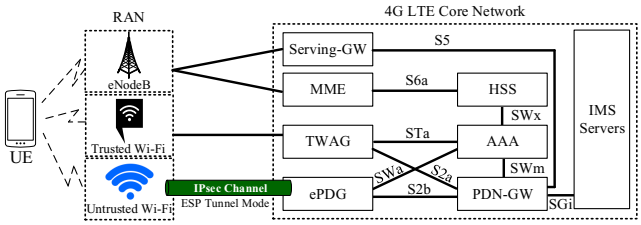


Fig. 1. The 4G LTE network architecture that supports the Wi-Fi calling service [9].

The rest of the paper is organized as follows. Section 2 presents the background of the Wi-Fi calling technology. Section 3 describes the threat model, methodology, and ethical considerations of this present study. Section 4 discloses the Wi-Fi calling vulnerabilities. Sections 5 and 6 present and evaluate two proof-of-concept attacks, namely the THDoS and user privacy leakage attacks, respectively. We propose a solution and evaluate it in Section 7. Section 8 presents related work, and Section 9 concludes this paper.

2 Wi-Fi Calling Primer

In this section, we introduce the network architecture and the voice call flow of the Wi-Fi calling services.

Network architecture: Figure 1 illustrates a simplified network architecture that supports both the Wi-Fi calling and VoLTE services. The UE (User Equipment), where the Wi-Fi calling and VoLTE applications are installed, connects to the similar network infrastructure including the RAN (Radio Access Network) and the CN (Core Network). For the RAN, VoLTE and Wi-Fi calling employ the eNodeB (Evolved Node B) and the Wi-Fi network, respectively. The 3GPP standard [18] classifies the Wi-Fi network into two types, namely *trusted* and *non-trusted*. For a cellular network operator, the

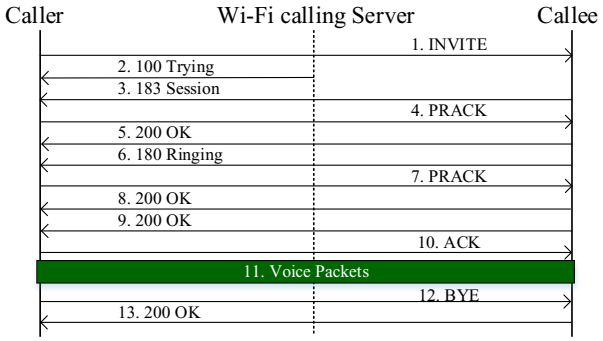


Fig. 2. Wi-Fi calling call flow diagram.

Wi-Fi networks deployed by itself are considered trusted, whereas the others are non-trusted.

The CN consists of eight main components: the S-GW (Serving Gateway), the PDN-GW (Public Data Network Gateway), the IMS (IP Multimedia Subsystem) servers, the TWGA (Trusted Wireless Access Gateway), the ePDG (Evolved Packet Data Gateway), the HSS (Home Subscriber Server), the MME (Mobility Management Entity), and the AAA (Authentication, Authorization, and Authorization) server. For the IMS traffic delivered between the UE and the IMS servers, the VoLTE packets are routed by the S-GW and the PDN-GW; those of Wi-Fi calling are routed by the trusted Wi-Fi network, the TWAG, and the PDN-GW, or by the untrusted Wi-Fi network, the ePDG, and the PDN-GW. The IMS servers offer multimedia services such as voice and text services in the cellular network. The HSS stores user subscription data while, together with the AAA, providing the user authentication service. The MME takes care of user mobility and network resource reservation.

In order to protect the UE and the CN from the access of the non-trusted Wi-Fi network, the Wi-Fi calling standard [1] stipulates that the UE and the CN shall support the EAP-AKA (Extensible Authentication Protocol - Authentication and Key Agreement) procedure [19], IKEv2 (Internet Key Exchange version 2), and IPSec [20]. Specifically, they have to authenticate each other based on the EAP-AKA procedure and then establish a secure IPSec channel using the ESP tunnel mode [21], [22] between the UE and the ePDG for the Wi-Fi calling services.

Wi-Fi calling call flow: Figure 2 shows the normal call flow of Wi-Fi calling. To initiate a call, the caller sends an SIP `INVITE` message, which specifies the capabilities (e.g., voice codec) of the caller, to the callee. Afterwards, the Wi-Fi calling server at the IMS system replies to the caller with an `100 Trying` message, which indicates that the call setup is in progress. In the meantime, the callee replies to the caller with a list of available voice codecs in an `183 Session` message. After receiving the message, the caller sends a `PRACK` (Provisional Acknowledgement) message to inform the callee of the selected codec. Once the `PRACK` is received, the callee phone starts to ring while sending back an `180 Ringing` message. The caller phone rings upon the arrival of the `180 Ringing` message. Whenever the callee answers the call, two call ends start to exchange voice packets for the voice call after the `200 OK` and `ACK` messages. A `BYE` message is sent from the end who terminates the call, and then the other end acknowledges it with a `200 OK` message.

3 Threat Model, Methodology and Ethical Considerations

Threat model: Compared to the limited deployment of trusted Wi-Fi networks, the non-trusted public Wi-Fi networks have been broadly deployed in practice, including those in campuses, libraries, grocery stores, coffee shops, to name a few. The present study mainly targets the security threats while users are using non-trusted public Wi-Fi networks. Adversaries are people or organizations which attack the Wi-Fi calling users. We consider the adversaries with the following capabilities: (1) they can *intercept*, *modify*, or *inject* any messages in the public communication channels (inside or outside connected Wi-Fi networks, e.g., Internet); (2) they adhere to all cryptographic assumptions, e.g., adversaries cannot decrypt an encrypted message without the decryption key; (3) they cannot compromise the Wi-Fi calling devices or the cellular network infrastructure, but may access/deploy surveillance cameras near the victims.

Methodology: We validate the vulnerabilities and the attacks on three major U.S. carriers, which together take about 75% of market share, and two Taiwan carriers, which together take 45% of market share. We conduct experiments using two Wi-Fi APs, a software-based AP based on a MacBook Pro 2014 laptop and an ASUS RT-AC1900 AP, and eight popular smartphones with the Wi-Fi calling service, which include Samsung Galaxy S6/S7/S8/J7, Apple iPhone6/iPhone7/iPhone8, and Google Nexus 6P. Apple and Samsung already take 74% share of the smartphone market [23]. The experiments are conducted in the Wi-Fi networks of several campuses, including Michigan State University, New York University, University of California Berkeley, and Northeastern University.

Ethical considerations: We understand that some feasibility tests and attack evaluations might be harmful to the operators and/or users. Accordingly, we proceed with this study in a responsible manner by running experiments in fully controlled environments. In all the experiments, victims are always our lab members. Our goal is to disclose new security vulnerabilities and provide effective solutions, instead of aggravating the damages.

4 Security Vulnerabilities of Wi-Fi Calling

In this section, we first introduce three security vulnerabilities discovered from operational Wi-Fi calling services in the U.S., and then present a study on non-U.S. operators and a feedback from the industry.

4.1 V1: WLAN selection mechanisms for Wi-Fi calling devices merely consider radio/connectivity capabilities of available Wi-Fi networks

The first vulnerability is that all studied Wi-Fi calling devices cannot exclude an insecure Wi-Fi network while enabling Wi-Fi calling services. According to Wi-Fi calling standards [1], [8], there are two Wi-Fi network selection modes: manual and automatic modes. In the manual mode,

No.	Time	Source	Destination	Protocol	Length	Info
440	56.276919	208.54.16.4	192.168.2.5	ESP	176	ESP (SPI=0xbb21253b)
441	56.266969	208.54.16.4	192.168.2.5	ESP	176	ESP (SPI=0xbb21253b)
465	56.316883	192.168.2.5	208.54.16.4	ESP	176	ESP (SPI=0x0855c9c8)
468	56.337334	192.168.2.5	208.54.16.4	ESP	176	ESP (SPI=0x0855c9c8)
469	56.347763	208.54.16.4	192.168.2.5	ESP	176	ESP (SPI=0xbb21253b)
470	56.348012	208.54.16.4	192.168.2.5	ESP	176	ESP (SPI=0xbb21253b)

Fig. 3. A trace of the Wi-Fi calling packets intercepted based on the ARP spoofing.

devices maintain a prioritized list of selected Wi-Fi networks, the implementation of which is vendor-specific. In the automatic mode, devices select their connected Wi-Fi networks by following the guidance from the network infrastructure based on the ANDSF (Access Network Discovery and Selection Function) procedure described in [9]. However, both modes do not consider security risks of available Wi-Fi networks but radio quality (e.g., ThreshBeaconRSSIWLANLow [8]) and connectivity capabilities, such as MaximumBSSLoad (i.e., the loading of Wi-Fi AP), MinimumBackhaulThreshold (e.g., 2 Mbps in the downlink) [9], [24].

Validation: We deploy two Wi-Fi routers of the same model to test the Wi-Fi network selection of the Wi-Fi calling devices. The experiment is conducted with four steps as follows. First, those two routers are deployed 5 and 10 meters, respectively, away from the tested devices. All test Wi-Fi calling devices are pre-installed with the required credentials to access these two Wi-Fi routers. Second, the security mechanism against the ARP (Address Resolution Protocol) spoofing attack, which is the prerequisite of various MitM (Man-in-the-Middle) attacks, is enabled on the far router, but it is disabled on the near router. Third, we launch an ARP spoofing attack from a computer that connects to the near router, to perform an MitM attack against all the other devices connecting to the router. Fourth, we enable the Wi-Fi calling service on all the tested devices, and then make a Wi-Fi calling call on each device whenever the device successfully has a Wi-Fi network connected.

We have three observations from the experiment. First, all the test Wi-Fi calling devices connect to the near Wi-Fi router. Second, all the Wi-Fi calling packets from the tested devices are intercepted by the computer based on the ARP spoofing attack, as shown in Figure 3. Third, none of the tested devices disconnects from the near router or terminates their Wi-Fi calling services; not any alerts or warnings are observed from the tested devices. This validation experiment confirms that current WLAN selection mechanisms do not prevent the Wi-Fi calling devices from connecting to an insecure Wi-Fi network, thereby causing them to suffer from the MitM attack. Note that the MitM attack does not need to compromise or control the near router.

Security implications: It is not without reasons that the WLAN selection mechanisms do not take security issues into consideration but consider only the radio quality or/and WLAN performance, since the Wi-Fi calling sessions have been protected by the IPSec tunnels with the end-to-end confidentiality and integrity protection. Although the security protection can prevent the Wi-Fi calling packets from being decrypted or altered, intercepting or discarding those packets for further attacks is still possible. We believe

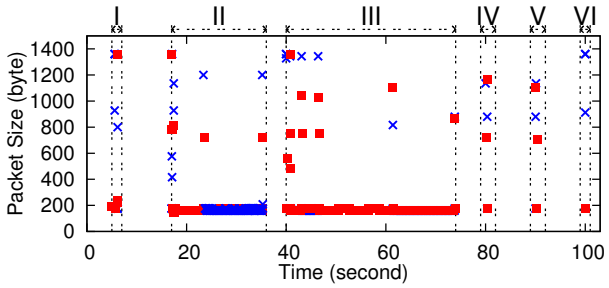


Fig. 4. The IPSec packets of six Wi-Fi calling events over time (×: uplink packets; ■: downlink packets; I/VI: Activating/Deactivating Wi-Fi calling; II/III: Receiving/Dialing a call; IV/V: Sending/Receiving a text).

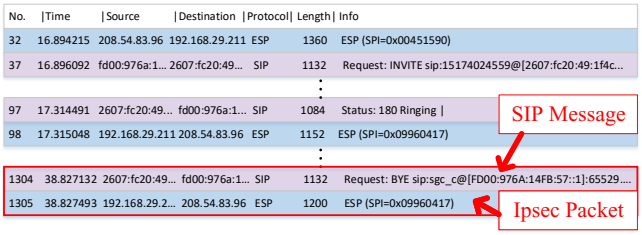


Fig. 5. A trace of the Wi-Fi calling packets collected on a test phone: SIP and IPSec packets.

that 3GPP and GSMA shall revisit the Wi-Fi network selection mechanisms for the Wi-Fi calling service in terms of security; otherwise, the Wi-Fi calling users are being exposed to potential security threats.

4.2 V2: Potential Side-channel Inference

Given the security mechanisms of untrusted access, the packets of the cellular services under untrusted Wi-Fi networks can be securely delivered through the IPSec channel between the UE and the ePDG. However, we discover that for all the test operators, the Wi-Fi calling service is the only service carried by the IPSec channel. This monotonous operation may allow the adversary to monitor the channel and then launch a side-channel attack to infer user privacy from the Wi-Fi calling events (e.g., call and text messaging statuses) and call statistics.

Validation: We examine whether any information can be inferred based on the intercepted Wi-Fi calling packets, which are encrypted by IPSec. After analyzing their patterns, we discover that for all the three operators, there are six service events of the Wi-Fi calling service, namely dialing/receiving a call, sending/receiving a text message, and activating/deactivating the service.

Figure 4 shows the IPSec packets captured on a Wi-Fi AP when the above six events are triggered on a test phone connecting to the AP. It is observed that all the events differ from each other in terms of traffic patterns, which are composed of packet direction (uplink or downlink), packet size, and packet interval. In order to automatically identify them based on the encrypted Wi-Fi traffic, we apply a decision tree method, the C4.5 algorithm [25]. To prepare a set of training data, we trigger those six events on the test phone with 50 runs each while collecting all the IPSec packets on the Wi-Fi AP. Based on the training data, a classification model can be generated by the C4.5 algorithm. We assess the classification accuracy of the model using 50 tests by

Test Device	US-I	US-II	US-III
Samsung J7 (US-III)	N/A	N/A	100%
Samsung S6 (US-II)	N/A	100%	N/A
Samsung S7 (US-I)	100%	N/A	N/A
Samsung S8 (US-II)	N/A	100%	N/A
Nexus 6P	100%	N/A	N/A
iPhone 6	100%	100%	100%
iPhone 7	100%	100%	100%
iPhone 8	100%	100%	100%

TABLE 2
Classification accuracy of the Wi-Fi calling events in various cross-phone and cross-carrier cases. N/A means that the test phone does not support the carrier's Wi-Fi calling service.

comparing the model's output with the test phone's packet trace as shown in Figure 5. The result shows that the model can give 100% accuracy. Note that the test phone is Nexus 6P with the Wi-Fi calling service of US-I.

We next examine whether the classification model works for cross-phone and cross-carrier cases. We consider various devices with the Wi-Fi calling services of the three carriers. Table 2 summarizes the result. It is observed that those six events in all the test cases can be identified accurately. Accordingly, the model that is derived based on the training data collected from Nexus 6P with the US-I's Wi-Fi calling service can be applied to the other devices and carriers, which include the Samsung Galaxy J7/S6/S7/S8 and iPhone 6/7/8 devices with the US-II/US-III networks.

Security implications: The IPSec channel can prevent man-in-the-middle attackers from decrypting or altering the Wi-Fi calling packets, but does not block the side-channel inference attack. Its monotonous operation allows the adversary to collect '*clean*' Wi-Fi calling traffic, which simplifies the side-channel inference.

4.3 V3: the Inter-system Service Continuity Mechanism of Wi-Fi Calling can be Bypassed

The inter-system service continuity mechanism can seamlessly switch the voice service of Wi-Fi calling on a device back to the cellular-based voice service (e.g., VoLTE), when the device disconnects from its connected Wi-Fi network or it cannot be reached through the Wi-Fi network (e.g., no response from the device in the Wi-Fi calling service). The mechanism can be triggered by the device or the cellular network infrastructure, and mainly consists of two steps, namely an inter-system handover [9] between Wi-Fi and the cellular network, and a procedure of the IMS service continuity [2]. Its operation can inherently protect the device against a DoS attack on the Wi-Fi calling service. For example, when all the Wi-Fi calling packets are maliciously dropped, the device is unreachable. However, the operation is not bullet-proof and may be bypassed with a sophisticated attack.

Validation: We conduct experiments to examine whether the mechanism can be bypassed in any scenarios. We test a Wi-Fi calling device with the following four scenarios, together with their corresponding results. First, the device with an established voice call of Wi-Fi calling moves out of its connected Wi-Fi network. We observe that the ongoing

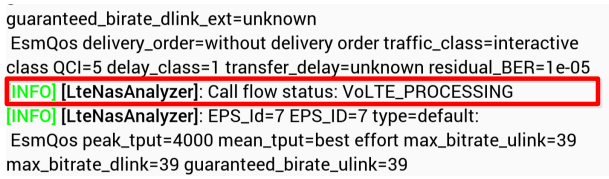


Fig. 6. A trace shows that a device switches an ongoing call attempt from Wi-Fi calling to VoLTE after all the Wi-Fi calling packets are dropped. It is obtained on the test phone via the software MobileInsight [26].

voice call can successfully migrate from Wi-Fi calling to VoLTE without any call interruption. Second, the device is dialing a Wi-Fi calling call while all its Wi-Fi calling packets are discarded from the Wi-Fi AP. We find that the device successively sends a packet of `SIP INVITE` to the Wi-Fi calling server; after six attempts, it switches to initiating a VoLTE call, as shown in Figure 6. Third, while the device is having an incoming call, all the Wi-Fi calling packets are discarded. It is observed that the device switches to VoLTE for the incoming call. Fourth, the packets of a Wi-Fi calling call on the device are discarded right after the call is established. We observe that no voice can be heard from two call ends, but the inter-system switch is not triggered and the device keeps the connection of the Wi-Fi network.

In summary, the inter-system service continuity mechanism is triggered only when the radio quality of the connected Wi-Fi network becomes bad, or the device and the network infrastructure cannot reach each other in the Wi-Fi calling service. As in the above fourth case, where the device and the network can reach each other but some packets are dropped, the adversary can attack a device's Wi-Fi calling call while keeping the device using the Wi-Fi calling service by preventing the inter-system switch from being triggered.

Security implications: Although the Wi-Fi calling standard [1], [2], [9] provides the inter-system switch mechanism for the Wi-Fi calling service continuity, it may suffer from some sophisticated attacks where the Wi-Fi calling packets can be intercepted. The interception is possible since the Wi-Fi calling traffic needs to traverse untrusted Wi-Fi networks and the Internet. To prevent the attacks, the service continuity mechanism should also take security concerns into consideration.

4.4 A Vulnerability Study on Non-U.S. Operators

We conduct a study of the Wi-Fi calling vulnerabilities on two Taiwan operators to examine whether they are limited to only U.S. operators or not. We summarize the result of the test phone, Samsung Galaxy S8, for each vulnerability as follows.

V1: We repeat the validation experiment of V1 on the phone with the Taiwan operators, and observe the same result that the WLAN selection mechanism does not prevent the device from connecting to an insecure Wi-Fi network, where an ARP spoofing attack is launched.

V2: For both Taiwan operators, we observe that the Wi-Fi calling service is also the only one service carried by the IPSec channel, and then apply the same classification method described in Section 4.2 into classifying the aforementioned six events. The result summarized in Table 3

Operator	Act./Deact. Wi-Fi calling	Rec./Dialing a call	Sending/Rec. a text
TW-I	100%/100%	100%/100%	N/A
TW-II	100%/100%	100%/100%	100%/100%

TABLE 3
Classification accuracy of the six Wi-Fi calling events for two Taiwan operators. N/A means that the event is not supported.

confirms that the method can give 100% accuracy for the event inference.

V3: We test the device with the Wi-Fi calling services of the Taiwan operators for the inter-system service continuity mechanism. It is also observed that the mechanism is deployed and can be bypassed in the fourth scenario described in Section 4.3.

4.5 Industry Feedback

We have reported the vulnerabilities to the U.S. operators that are studied in this work and several device manufacturers including Google, Samsung, and Apple. In particular, the Google Android security team gives a positive feedback that the team has confirmed our findings after a security analysis of the vulnerabilities, and will address them in an upcoming security patch. We thus received a Google Security Reward in Jan. 2020. On the other hand, we are awaiting hearing from the other operators and manufacturers.

5 Telephony Harassment/Denial of Voice Service (THDoS) Attack

We next devise the THDoS attack, which can cause telephony harassment or denial of voice service on the Wi-Fi calling users. In the following, we describe the attack design, evaluation and real-world impact.

5.1 Attack Design

In this attack, the adversary manages to discard particular signaling or/and voice packets of Wi-Fi calling from the victim device, while preventing the inter-system service continuity mechanism from being triggered. The attack can cause damage on the device's voice service supported by Wi-Fi calling, and let the damage last by getting the device stuck with the Wi-Fi calling service. To discard particular packets between the device and the network infrastructure, the adversary needs to identify encrypted IPSec packets. We next start with an illustrative example of the Wi-Fi calling call, and then analyze the traffic patterns of the Wi-Fi calling messages and events based on the encrypted packets.

An illustrative example: A device user receives an incoming Wi-Fi calling call and answers it around 6 seconds after its ringtone. Afterwards, the user has a voice conversation for around 12 seconds. Figure 7 shows the IPSec packets observed on the Wi-Fi AP to which the device connects. The following four events can be observed: (1) receiving a call with a ringtone; (2) answering a call; (3) talking; (4) hanging up a call.

Event 1: Receiving a call with a ringtone. Figures 8(a) and 8(b) show the downlink and uplink packets of this event, respectively. The first incoming packet, which is intercepted at the 2nd second, is a 1360-byte IPSec packet. We

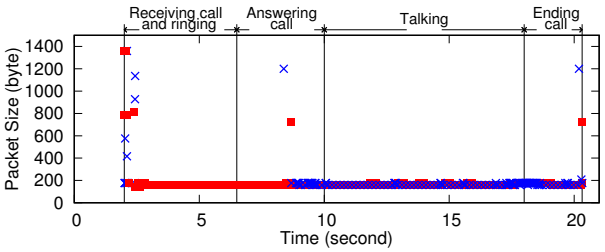
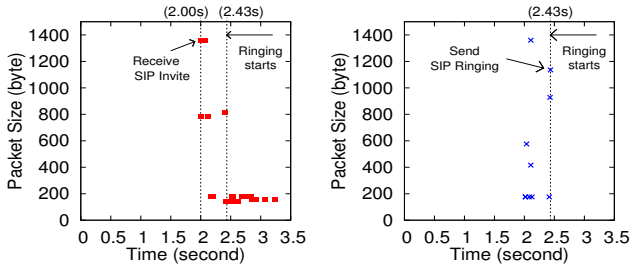


Fig. 7. The IPSec packets of a Wi-Fi calling call, which are observed on the Wi-Fi AP to which the callee connects. (×: uplink packets; ■: downlink packets).



(a) Downlink (sent by the server) (b) Uplink (sent by the callee)

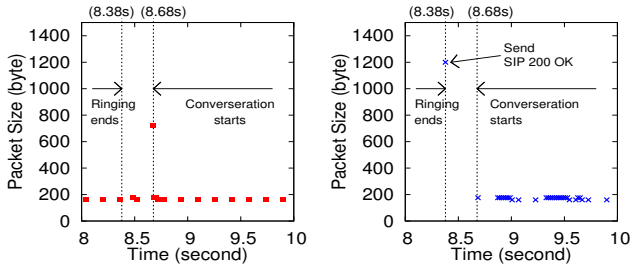
Fig. 8. The packet arrivals of the event 'receiving a call with a ringtone' on the Wi-Fi AP.

decrypt it at the callee and identify it as an SIP `INVITE` message, which indicates that a call attempt is coming. At the 2.43th second, the callee sends an `180 RINGING` message to the Wi-Fi calling server. Afterwards, it is observed that several small IPSec packets with only 176 bytes are received by the callee, but the callee does not send any packets back. We discover that they are voice packets in the RTP (Real-Time Protocol) protocol.

Event 2: Answering a call. As shown in Figures 9(a) and 9(b), the callee answers the call at the 8.38th second by sending a `200 OK` message to the server, and then receives an acknowledgment at the 8.68th second. Afterwards, the call conversation starts and the callee begins to send/receive voice packets.

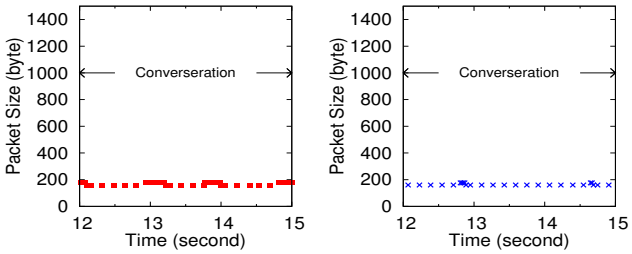
Event 3: Talking. The traffic pattern of this event is shown in Figure 10. During the call conversation, the callee keeps sending/receiving voice packets to/from the Wi-Fi calling server, but no SIP messages are observed. We further discover that the callee at least receives 10 voice packets every two seconds from the server.

Event 4: Hanging up a call. The callee sends a `BYE` message at the 20.19th second after the call is hanged up, as shown



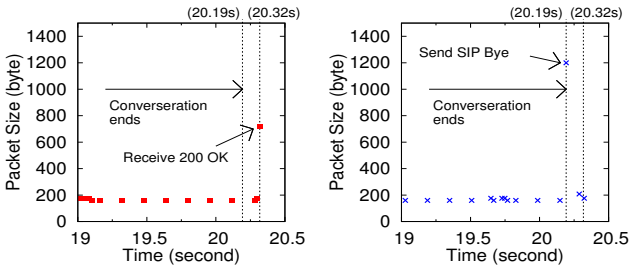
(a) Downlink (sent by the server) (b) Uplink (sent by the callee)

Fig. 9. The packet arrivals of the event 'answering a call' on the Wi-Fi AP.



(a) Downlink (sent by the server) (b) Uplink (sent by the callee)

Fig. 10. The packet arrivals of the event 'talking' on the Wi-Fi AP.



(a) Downlink (sent by the server) (b) Uplink (sent by the callee)

Fig. 11. The packet arrivals of the event 'hanging up a call' on the Wi-Fi AP.

No.	Dropped Packets	Sender	Results
1	INVITE	Caller	Caller initiates a cellular-based call.
2	100 Trying	Server	No effect.
3	183 Session	Callee	Two outgoing calls arrive at callee.
4	PRACK	Caller	No effect.
5	200 OK	Callee	No effect.
6	180 Ringing	Callee	Caller will not enter the conservation state. The caller phone gets stuck in the dialing screen.
7	PRACK	Caller	No effect.
8	200 OK	Callee	Caller keeps hearing the alerting tone.
9	200 OK	Callee	Caller keeps hearing the alerting tone.
10	ACK	Caller	No effect.
11	Voice Packets	Caller/Callee	Call drops or voice quality downgrades.
12	BYE	Caller	Callee gets stuck in the conversation state for 20 s. Afterwards, the call is terminated.
13	200 OK	Callee	No effect.

TABLE 4
The results obtained when we drop different patterns of the signaling and voice packets for an outgoing Wi-Fi calling call.

Drop Rate (%)	Voice Quality
below 20%	No clear impact.
40-60%	Some noises.
70-90%	Conversation is hardly continued.
100%	Call is terminated by the network.

TABLE 5
Voice quality varies with the drop rate of voice packets.

in Figure 11(b). After the 20.32nd second, no more IPSec packets are observed. Note that if the caller hangs up the call first, the `BYE` message should be sent by the server.

Traffic Pattern Analysis: We have five observations on the traffic patterns of the Wi-Fi calling messages and events.

1) The sizes of the voice packets in IPSec are smaller than 200 bytes (e.g., 176 bytes).
2) The sizes of the SIP packets that contain signaling messages, including `INVITE`, `180 RINGING`, `200 OK`, and `BYE`), in IPSec are much larger than the voice packets (e.g., 800-1360 bytes).
3) The callee starts to receive the voice packets from the Wi-Fi calling server after the `180 RINGING` message is sent.
4) No voice packets are sent out by the callee before the call conversation starts.
5) The callee keeps receiving more than 10 voice packets every two seconds from the Wi-Fi calling server after the call conversation starts.

These patterns allow us to identify call events, e.g., an outgoing call is initiated, an incoming call attempt arrives, and an ongoing call ends. Moreover, by correlating them with the call flow of Wi-Fi calling (see Figure 2), the signaling messages of Wi-Fi calling can be identified purely based on the encrypted IPSec packets. Note that the third observation is made only from US-I and US-II; the others can be observed from all the test operators.

5.2 Attack Evaluation

We launch attacks by discarding different patterns of the signaling and voice packets for an outgoing call of Wi-Fi calling. Table 4 summarizes the results, which are observed on all the tested smartphones. We exploit the results to devise four attack variants as follows. Note that the damage that is caused to mobile phones may not be applied to other SIP phones (e.g., Cisco SPA 525G2).

Annoying-Incoming-Call Attack: The callee as the victim would receive multiple incoming calls from the caller. There are two approaches. First, the adversary drops the `183 Session Progress` message sent by the callee, and then the caller's Wi-Fi calling device would initiate another VoLTE call towards the callee. Second, the adversary discards the `180 Ringing` message sent by the callee, and then it would cause the caller's Wi-Fi calling device to get stuck in the dialing screen. The caller does not hear any alerting tone, but the callee's device would ring. The caller may thus keep redialling.

Zombie-Call Attack: The caller's device can be forced to get stuck in the dialing screen, when the adversary discards the `200 OK` message sent by the callee. The message indicates that the call has been answered, so without receiving the message, the caller's device gets stuck in the dialing screen and keeps hearing the alerting tone. The call conversation is thus never started.

Intermittent Mute Call Attack: Two parties of a Wi-Fi calling call are both victims. This attack does not aim to terminate the call but only mute the victims' voice for a certain time. Our result shows that the adversary can mute the call up to 8 seconds by dropping voice packets. If the voice suspension time is longer than 8 seconds, the call would be terminated by the network. To prolong the attack period, the adversary can launch a cyclical attack that drops voice packets for 7 s and skip the packets for the next 1 s to mute the call intermittently.

Telephony Denial-of-Voice-Service Attack: Both the caller and the callee are victims. This attack downgrades the voice quality of a Wi-Fi calling call so that the conversation is hard to be continued; meanwhile, the inter-system service continuity mechanism is not triggered. It is achieved by controlling the drop rate of the intercepted Wi-Fi calling

packets to/from the victim. Table 5 shows the negative impact on the voice quality with different drop rates. There are four findings. First, when the drop rate is below 20%, the caller/callee users do not complain about any voice quality downgrade. Second, when the drop rate increases to 40%-60%, some of the users may notice some noises. Third, when the drop rate becomes 70%-90%, the voice call is hardly continued. Fourth, when the drop rate is 100%, the call is terminated within 8 seconds. Note that when the drop rate is below 90%, the call termination is never triggered.

5.3 Real-world Impact

The impact of the THDoS attack can be significant in practice. Our studies show that the campus Wi-Fi networks, which most U.S. universities have deployed, are the best attack surfaces for the adversary. For example, the campus Wi-Fi (MSUNet) in Michigan State University provides students, the faculty, and the staff with free Wi-Fi access. In a 2-min experiment, we discover that more than 700 devices including smartphones, tablets, and computers, connect to MSUNet. All the devices are served by the same gateway which is vulnerable to an ARP spoofing attack, so their Wi-Fi calling packets can be intercepted if there are any. Therefore, it allows the adversary to launch the THDoS attack against the Wi-Fi calling devices under the gateway. Note that MSUNet is not the only Wi-Fi infrastructure that suffers from the ARP spoofing and THDoS attacks. We find that such vulnerability also exists in the campus Wi-Fi of many other universities, such as New York University, University of California Berkeley, Northeastern University, etc.

6 User Privacy Leakage Attack

In this section, we devise a proof-of-concept attack that can leak the privacy of the Wi-Fi calling users. We exploit the discovered vulnerabilities to collect call statistics (e.g., call duration and number of dialing calls) for each Wi-Fi calling device with a specific IP address in an area, while using the nearby cameras to identify the person behaviors related to phone usages. By considering two information sources together, a device's call statistics can be correlated with a person's behavior. For example, a device with 5-second call duration can be correlated with a person who holds his/her phone and speaks for 5 seconds. Based on such correlation, the adversary can obtain the IP address of a specific Wi-Fi calling user and then identify the user's packets. The adversary can thus inspect the packets to infer the user's privacy, including device activities (e.g., accessing gmail), device information (e.g., iPhone 7), running applications (e.g., WeChat), etc. In addition, several prior studies have demonstrated that the call statistics can be exploited to infer some user privacy information including mood (e.g., stressful [13]), personality (e.g., conscientiousness [12]), malicious behaviors (e.g., dialing spamming calls) [14], to name a few.

6.1 Overview of Attack Design

We launch this attack by developing a user privacy inference system, called UPIS, as shown in Figure 12. It consists of three major components: WiCA (Wi-Fi Calling Analyzer), UCIA (User Call and ID Analyzer), and CS-IP2U (Call Statistics based IP-to-User correlation) modules. WiCA intercepts all the Wi-Fi packets and then identifies the Wi-Fi calling ones. From the Wi-Fi calling packets, WiCA extracts call statistics (e.g., ringing time and call duration) for each device IP. The other packets are dispatched to a real-time traffic analyzer, which analyzes application identity and device information. UCIA identifies each phone user's call statistics based on a surveillance camera using the techniques of face recognition and human motion detection. CS-IP2U uses the call statistics from both WiCA and UCIA to correlate each phone user with an IP address. It generates a mapping table with IP and user identity, together with each user's call statistics. We next elaborate on the WiCA, UCIA, and CS-IP2U components, and finally evaluate the UPIS system.

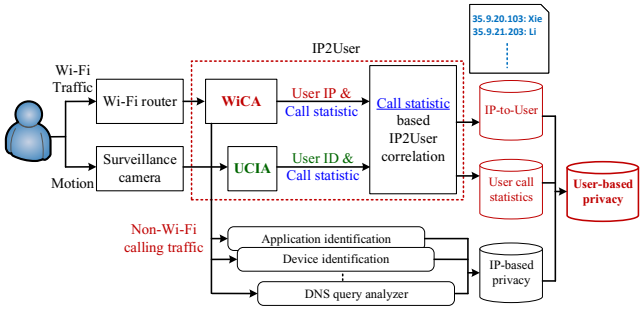


Fig. 12. The UPIS system that infers user privacy of the Wi-Fi calling users.

6.2 WiCA: Wi-Fi Calling Analyzer

WiCA infers call statistics on a per-IP basis by analyzing the Wi-Fi calling traffic. Unlike the aforementioned THDoS attack where specific signaling messages of Wi-Fi calling need to be accurately identified, WiCA considers the extraction of only call statistics. Thus, it requires a relatively simple approach that consumes little resources. Figure 13 illustrates its finite state machine, where the initial state is `IDLE`. It works as follows.

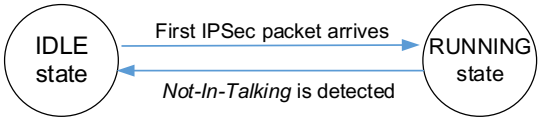


Fig. 13. The state transition diagram of WiCA.

Step 1: At the initial `IDLE` state, whenever any IPSec packet belonging to a call event is received, WiCA moves to the `RUNNING` state. WiCA determines that kind of IPSec packets by checking whether they are sent to/from any Wi-Fi calling servers. WiCA records the forwarding direction to differentiate between two events, namely *dialing a call* and *receiving a call*. Their IPSec packets are sent to and from the servers, respectively.

Step 2: At the `RUNNING` state, WiCA uses a 2-second time window to group the collected IPSec packets and classifies them into three categories: *C-Large*, *C-Middle*, and *C-Small*. They include the packets with the sizes larger than 800 bytes, between 200 and 800 bytes, and smaller than 200 bytes, respectively. The *C-Large* category includes some

Conditions		Identified Scenarios
$Num\_UL\_C_{Small}$	$Num\_DL\_C_{Small}$	
=0	>10	Ringing
>10	>10	Talking
=0	=0	Not in Talking

TABLE 6
$Num\_UL\_C_{Small}$ and $Num\_DL\_C_{Small}$, which respectively represent numbers of uplink and downlink packets smaller than 200 bytes within 2 seconds, are used to determine *Ringing*, *Talking*, *Not in Talking* scenarios for US-I, US-II, US-III. Note that the rule of determining ringing event is only applicable to US-I and US-II but not to US-III, since which US-III does not send small voice packets to the Wi-Fi calling callee when his/her phone is ringing

critical SIP call messages (e.g., `INVITE` and `RINGING`), whereas the *C-Small* contains voice packets. Note that the 2-second packet collection is denoted as $Data_{2sec}(x)$, where x is the sequence of a series of the 2-second collection sets.

Step 3: WiCA identifies three scenarios, namely *Ringing*, *Talking* and *Not in Talking*, based on the number of uplink and downlink *C-Small* packets in $Data_{2sec}(x)$, which are denoted as $Num\_UL\_C_{Small}$ and $Num\_DL\_C_{Small}$, respectively. The rules are summarized in Table 6. When no event is identified in a collection set, $Data_{2sec}(x)$, it is buffered and WiCA moves back to `Step 2`. When any event is identified, WiCA takes subsequent actions for the event in the following.

- *Ringing:* WiCA revisits the last collection, $Data_{2sec}(x-1)$, and looks for the time when the last *C-Large* IPSec packet is captured, which is considered as when the ring starts. We denote the time as $T_{RingingStart}$.
- *Talking:* When no *Talking* scenarios are identified before this scenario, WiCA revisits the last collection, $Data_{2sec}(x-1)$, and finds the time when the first *C-Large* IPSec packet (i.e., `SIP 200 OK`, which indicates the event 'answering the call') is captured. This time, denoted as $T_{TalkingStart}$, is considered as the time when the talk starts.
- *Not In Talking:* WiCA revisits the last collection, $Data_{2sec}(x-1)$, to discover the time when the first *C-Large* IPSec packet (i.e., `SIP BYE`) is captured. This time, denoted as $T_{CallEnd}$, is considered as the time when the call ends. When the *C-Large* packet is sent by the Wi-Fi calling device, WiCA infers that the device user hangs up the call first. Otherwise, the other call end terminates the call first. When the call termination is observed, a pattern analyzer outputs a set of information including the call end initiating the call, ringing duration (i.e., $T_{TalkingStart} - T_{RingingStart}$ or $T_{CallEnd} - T_{RingingStart}$), talking duration (i.e., $T_{CallStop}$-$T_{TalkingStart}$), and the call end terminating the call. Afterwards, WiCA returns to the `IDLE` state. Note that the talking duration is not applicable to unestablished calls.

6.3 UCIA: User Call and ID Analyzer

UCIA is a visual recognition system which identifies users and their motions related to making calls (e.g., a user moves a phone close to his/her ear). It mainly leverages four computer vision techniques including a tiny face detector, which is designed to find small faces in a video, DR-GAN (Disentangled Representation learning-Generative Adversarial Network), HOG (Histogram of Oriented Gradient) [27], and SVM (Support Vector Machine). UCIA does not require the users to be still or use a high-resolution video. It can support the video in which face resolutions are as low as 25x10 [28].

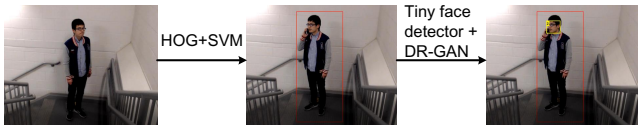


Fig. 14. The UCIA working flowchart. The red bounding box denotes a detected calling/talking motion, whereas the yellow bounding box denotes a detected user face.

Figure 14 illustrates the UCIA working flowchart, which analyzes videos on a per-frame basis. It consists of two modules: (1) calling/talking motion detection and (2) user face detection and recognition. In each video frame, UCIA uses the HOG and SVM models to detect calling/talking motions for all users, and labels those whose motions are detected using red bounding boxes. For each red bounding box, UCIA further uses the tiny face detector and the DR-GAN model to label the user face with a yellow bounding box, and identifies his/her identities (i.e., names). We next detail these two modules and then evaluate the performance of UCIA.

6.3.1 Calling/talking Motion Detection

UCIA generates features of target motions using the HOG descriptor, and then classify them with an SVM model.

SVM: We train the SVM model to recognize two motions, namely 'dialing a call' and 'talking in a call'. Since no video datasets contain them, we invite twenty volunteers to record videos of their dialing/talking motions. To differentiate those two motions from the others, we do the model training by mixing the recorded videos with those from 101 action categories in the UCF101 database [29].

HOG: Each frame in a surveillance video may contain many candidate bounding boxes within a sliding window. After all the persons are marked by the bounding boxes, the pre-trained SVM classifier determines whether any of those two motions happens in each bounding box based on the change of the gradient information described by the HOG descriptor. To implement the HOG descriptor, we first divide each image into different small connected components, called cells, and then collect the orientation histogram of gradients for each pixel within each cell. Finally, we concatenate all the histograms to be the HOG descriptor.

6.3.2 User Face Detection and Recognition

We adopt a tiny face detector [28], which is based on the technique of deep convolution neural network (CNN), to detect user faces, since not all the surveillance cameras offer high video quality (e.g., 1080p). The detector is designed to detect small faces (e.g., a face with the size of $3 \times 3cm^2$) in a low-resolution video, but can also support large faces in a high-resolution video. Moreover, since people do not always face to the cameras with a frontal view, extracting pose-invariant feature representations is critical to the face recognition. We thus apply DR-GAN [30] that can generate those representations to recognizing user identities.

Tiny face detector: The working flowchart of this detector is illustrated in Figure 15. The detector first resizes each input

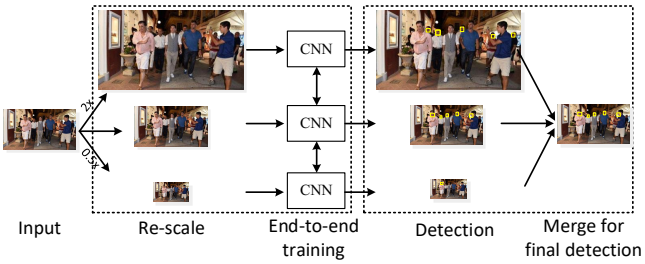


Fig. 15. The working flowchart of the tiny face detector.

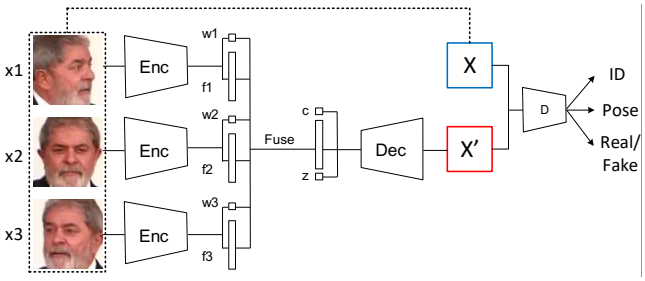


Fig. 16. Overview of the DR-GAN model.

image into other two images with different resolutions to construct an image pyramid for the training. It uses those three images with different resolutions as the input of the CNN model. We adopt a well-trained model provided by Hu et al. [28]. The trained model can be used to predict the bounding boxes on the image pyramid. All the detected bounding boxes are then selected and merged based on the non-maximum suppression (NMS) method [31], and then the final detection result on the original image can be obtained.

DR-GAN-based face recognition: To recognize user identities, it is challenging to deal with variations on the user faces (e.g., illumination conditions, poses, and expressions); especially, the pose changes can cause a big drop on the face recognition performance. We tackle this challenge by applying the DR-GAN model in the following two steps. First, we define face angles ranging from -90° to 90°. With the 0° face angle, the face is in the frontal view, which almost contains all the facial information. With the angle of -90° or 90°, only one side of the face is visible so that it is difficult for the model to do face recognition. Second, we leverage the DR-GAN model to extract the disentangled face representation by fine-tuning the GAN (Generative Adversarial Networks). The model can generate a representation for each face with personal identity information and then the representation can be used for the face verification and identification.

The face recognition flowchart of the DR-GAN model is shown in Figure 16. To train the DR-GAN model, several face images with different poses for the same user identity are used as the input. Each image will be fed into the encoder that uses VGG16 as the network structure. In addition to generating a 320-dim feature f for each face, the encoder outputs a 1-dim coefficient w. A fused feature f' can be then generated based on the following equation:

$$f' = \frac{\sum_i^n w_i f_i}{\sum_i^n w_i} \quad (1)$$

, where f' is a weighed average over all the f_i. f' can be fed into a decoder to generate an output image, called synthetic image, with the same size as the input. Accompanying the feeding of f', a pose code c and a random noise z are also appended. The former can help the decoder generate a synthetic image with an arbitrary pose, whereas the latter can prevent the decoder from overfitting. We further use the combination of the original face images and the synthetic image to train a discriminator. After the adversarial training involving the encoder, the decoder, and the discriminator converges, an updated encoder can be derived. We finally use this trained encoder to generate the disentangled feature representations of all the input images for the face recognition.

6.4 CS-IP2U: Correlating IP with User Identity

The CS-IP2U module correlates user identities with IP addresses based on the call statistics extracted by WiCA and UCIA. It mainly considers two kinds of events, namely call start and call end. We denote the happening times of these two events as $TCStart$ and $TCEnd$, respectively. Ideally, for an identified Wi-Fi calling call, WiCA outputs $TCStart_w$, $TCEnd_w$, and IP, whereas UCIA outputs $TCStart_u$, $TCEnd_u$, and $UserID$. One correlation can be thus identified when $TCStart_w == TCStart_u$ and $TCEnd_w == TCEnd_u$. Nevertheless, in practice, it is not the case due to the errors of recorded timing in the call statistics. CS-IP2U thus considers not only time points but also time intervals in the correlation with the following three steps.

Step 1: We consider two time intervals, $TCStartInt_w = [TCStart_w - \sigma, TCStart_w + \sigma]$ and $TCEndInt_w = [TCEnd_w - \sigma, TCEnd_w + \sigma]$, for the call start and end events in WiCA, respectively. σ is set to the maximum timing error observed in WiCA (i.e., 1 second).

Step 2: We further consider the other two time intervals, $TCStartInt_u = [TCStart_u - \epsilon, TCStart_u + \epsilon]$ and $TCEndInt_u = [TCEnd_u - \epsilon, TCEnd_u + \epsilon]$ for the call start and end events in UCIA, respectively. ϵ is set to the maximum timing error observed in UCIA (i.e., 1.5 seconds).

Step 3: When the following two conditions are met, the corresponding IP and $UserID$ are correlated: $TCStartInt_w \bigcap TCStartInt_u \neq \emptyset$ and $TCEndInt_w \bigcap TCEndInt_u \neq \emptyset$.

Note that current CS-IP2U implementation does not support the cases that multiple Wi-Fi calling users start or end calls near-simultaneously (within the time interval of $max\{\sigma, \epsilon\}$ (i.e., 1.5 seconds). To address this issue, more fine-grained call statistics should be extracted by the WiCA and UCIA modules. For example, we can infer time periods that users are talking and those that user are listening by analyzing the uplink and downlink Wi-Fi calling voice packets at the WiCA and detecting who are talking [32] at the UCIA. We do not implement this advanced feature on our attack prototype, but only demonstrate the feasibility of the correlation between user identities and IP addresses.

6.5 Attack Evaluation

We next evaluate the performance of the UPIS system in a controlled setting (in our laboratory without passersby)

and a wild setting (in an on-campus coffee shop with passersby). The WiCA is implemented using Python3 and the scikit-learn library [33] on a 2014 Macbook Pro laptop with a CPU, Intel I5-4278U, and an 8GB RAM. The UCIA is implemented using Python3 and other three computer vision and machine learning libraries, namely VLFeat [34], MatConvNet, and Tensorflow, on our campus computing servers (MSU HPCC) [35]. The CS-IP2U is also implemented on the Macbook laptop. Moreover, the CS-IP2U requires to associate time and events between the WiCA and the surveillance camera, so the clock synchronization between them is needed. The precision time protocol (PTP) [36] can be used for the synchronization.

6.5.1 Evaluation Metrics

WiCA: The evaluation metric is the estimation error of the call event time, which is the difference between the time when a W-Fi calling call starts or stops, and the time that is estimated for the call event by the WiCA. Note that we can use a command, `logcat -b radio -v threadtime | grep "update phone state"`, on Android phones to obtain the times of the call start and stop events, which are the ground truth in the evaluation.

UCIA: We evaluate UCIA from three aspects, namely calling/takling motion recognition, user identity recognition, and the estimation error of the call event time. The evaluation metrics of the first two aspects include accuracy (ACC), false positive rate (FPR), and false negative rate (FNR).

For the calling/takling motion recognition, the video frames of a user can be classified into two categories: with and without a calling event. They are considered as positive and negative cases, respectively. For the user identity recognition, UCIA analyzes all the frames that are recognized with a calling event and looks for the user identity in the event from our database. The ACC, FPR, and FNR rates are calculated on a per-user basis.

CS-IP2U: The evaluation metric is the ratio of the accurate cases that the identity of the Wi-Fi calling user is correctly correlated with the user's device IP, to all the user's Wi-Fi calling calls.

6.5.2 Experimental Results

We evaluate the performance of Wi-Fi calling user privacy inference system (UPIS) in the non-wild and wild settings as follows.

•**Using non-wild settings (without passersby):** The experiment is conducted in a on-campus space where there are no passersby but the experiment participants. We consider four participants in the experiment. In each test, each of them is requested to dial at least one call; they are allowed to do any random actions (e.g., looking at the ground). To emulate a real use scenario, we do not restrict the duration of each Wi-Fi calling call. The experiment includes 10 tests, and 10 videos are recorded.

The experimental result is summarized in Table 7. In the WiCA module, the errors of the call event time estimation are limited to at most 0.55 s. As for the UICA module, the ACC/FPR/FNR rates of the motion recognition are 85%∼94.9%, 3.4%∼9.1%, and 7.3%∼22.3%, respectively; those of the identity recognition are 92.5%∼98.1%, 1.0%∼7.5%, and 5.7%∼9.9%, respectively; the errors of the time estimation range between 0.53 s and 1.51 s. Although the identify recognition mechanism does not correctly recognize user identity in all the video frames, the 100% accuracy is not needed in practice. The reason is that the successful recognition of a Wi-Fi calling user requires only one video frame of the user. Lastly, the overall performance of the UPIS system is 97.33% (73/75) by considering the accuracy of the CS-IP2U module.

•**Using wild settings (with passersby):** We conduct the above experiment in an on-campus coffee shop where has not only experiment participants but also other customers. We compare the results of the wild experiment, which is also summarized in Table 7, with that of the controlled one. For the WiCA module, the performance is comparable to that of the controlled experiment. In the UICA module, the ACC/FPR/FNR rates of the motion recognition decrease to 80.1%∼88.3%, increase to 12.1%∼18.1%, and increase to 10.8%∼22.04%, respectively. This downgrade performance hurts the accuracy of the call event time estimation; thus, the combined error of the start and end times increases from 2.33 seconds in the controlled experiment to 2.89 seconds. The similar trends are also observed in the identity recognition; its ACC/FPR/FNR rates decrease to 90.8%∼93.8%, increase to 5.6%∼10.2%, and increase to 7.5%∼10.0%, respectively. As expected, the overall performance is reduced to 87% (66/76). The reason is that the unexpected passersby can affect the performance of the motion and identity recognition mechanisms. We leave the further improvement to our future work.

6.6 Real-world Impact

To the best of our knowledge, the UPIS is the first system which can correlate the identity of the Wi-Fi calling user with the user's device IP based on the call statistics of the Wi-Fi calling service. Seemingly, it needs a little strong assumption that the victims are in the visible area of a surveillance camera that can be accessed by the adversary and a face recognition technique can be applied. However, for the sake of public safety, such surveillance cameras with face recognition have been broadly deployed in several countries, e.g., United Kingdom [37], China [38], and U.S. (Chicago and Detroit) [39]. We thus believe that some use scenarios can benefit from the UPIS system in practice. For example, the UPIS can be deployed at airports to be against terrorists. It allows the law enforcement agents to identify suspects' phone models and IP addresses, and further remotely install the malware on their phones for monitoring. The remote installation can be achieved by exploiting public security vulnerabilities of the target devices. Note that we do not advocate any use scenarios compromising user privacy no matter whether the purpose is benign or not.

7 Solution: Wi-Fi Calling Guardian

To completely address all the identified vulnerabilities, it is required to modify current Wi-Fi calling standard; the standard modification is too time consuming to be achieved in a short time. We thus propose a software-based security framework, *Wi-Fi Calling Guardian*, to largely mitigate the

Module	Performance Metrics		Controlled settings				Wild settings			
			User1	User2	User3	User4	User1	User2	User3	User4
WiCA	Call Event Time Estimation	start time error (sec)	0.25	0.55	0.15	0.08	0.33	0.48	0.22	0.15
		end time error (sec)	0.17	0.23	0.37	0.10	0.27	0.38	0.54	0.31
UICA	Calling Motion Recognition	ACC	94.9%	90.1%	90.0%	85.0%	88.3%	85.7%	85.0%	80.1%
		FPR	3.4%	8.3%	6.8%	9.1%	12.7%	12.1%	13.1%	18.1%
		FNR	7.3%	12.5%	12.2%	22.3%	10.8%	17.6%	17.9%	22.04%
	Call Event Time Estimation	start time error (sec)	1.51	1.34	0.53	0.98	1.22	1.34	1.03	1.28
		end time error (sec)	0.62	0.99	0.76	1.19	1.23	1.55	0.92	1.26
	User Identity Recognition	ACC	95.8%	98.1%	92.5%	93.5%	91.3%	93.8%	92.0%	90.8%
		FPR	2.8%	1.0%	6.6%	7.5%	10.2%	8.0%	6.2%	5.6%
		FNR	8.3%	5.7%	9.9%	8.1%	7.6%	7.5%	10.0%	10.1%
CS-IP2U	ID and IP Mapping	ACC	95.0% (19/20)	100% (19/19)	100% (17/17)	94.7% (18/19)	83.3% (15/18)	84.2% (16/19)	89.4% (17/19)	90% (18/20)

TABLE 7
Overall performance of the UPIS system.

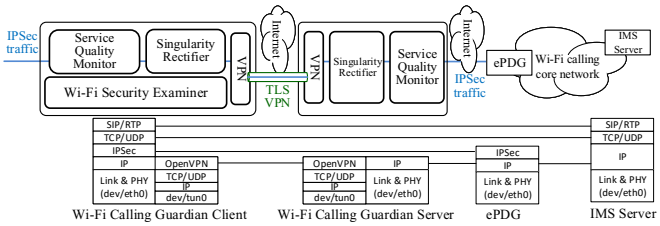


Fig. 17. The network architecture and protocol stack of Wi-Fi Calling Guardian.

impact of the vulnerabilities without any modifications on the standard but only a phone-side software upgrade. In the following, we present the design and evaluate its performance.

7.1 Design

The architecture of Wi-Fi Calling Guardian consists of two network elements, namely the client on the Wi-Fi calling device and the server in a secure private network, as shown in Figure 17. There are mainly three security modules on the client and the server: (1) *Wi-Fi security examiner*, which examines whether the connected Wi-Fi network is secure for the Wi-Fi calling service; (2) *singularity rectifier*, which introduces noises to mix with the Wi-Fi calling traffic, thereby increasing the difficulty of the inference; (3) *service quality monitor*, which monitors whether the Wi-Fi calling user is suffering from the degradation of the service quality and then takes actions if needed.

Ideally, the Wi-Fi security examiner can help the Wi-Fi calling device stay away from insecure Wi-Fi networks, which are vulnerable to any known attacks (e.g., the ARP spoofing attack). However, the situation is far from simple in practice due to three reasons. First, not all the vulnerabilities can be identified using a passive approach in which the examiner operates (e.g., using detection only not launching attacks). Second, the Wi-Fi calling user may have no secure Wi-Fi networks to associate with. Third, the proposed attacks (e.g., the THDoS and user privacy leakage attacks) can be launched outside of the connected Wi-Fi network. Therefore, the Wi-Fi security examiner uses a passive approach to explore the insecurity of the connected Wi-Fi network on one hand; on the other hand, the other security modules, *singularity rectifier* and *service quality monitor*, protect the Wi-Fi calling device against potential attacks. We next elaborate on the details of these three security modules.

Wi-Fi security examiner: Two detection mechanisms are deployed to examine the insecurity of the connected Wi-Fi network. First, this module detects whether the WPA3 protocol [40] is enabled in the connected Wi-Fi network. It is because the WPA3 requires all the compliant devices to support the PMF (Protected Management Frames) feature, which provides integrity protection over management frames and can thus defend against some Wi-Fi attacks (e.g., deauthentication and rogue AP attacks). Second, this module detects whether the Wi-Fi calling device is being under an ARP spoofing attack, which is a prerequisite of various MitM attacks, so that V1 can be prevented. It monitors the device's ARP table and checks whether two different IP addresses associate with the same MAC address.

Singularity rectifier: This module uses a *normalized data transmission* mechanism to prevent the Wi-Fi calling service from appearing as a singular service supported by the IPSec channel. The mechanism encapsulates all the Wi-Fi calling packets into UDP datagrams for the delivery. The UDP datagrams with a fixed packet size (e.g., 300 bytes) are generated by both the client and the server, and sent to the other end at a steady rate. This approach can remove two traffic patterns of Wi-Fi calling, namely *packet sizes* and *delivery directions*, at a low cost (e.g., consuming only the bandwidth of 0.032 MB/s while the rate is 50 pkts/s) so that V2 can be eliminated.

Service quality monitor: This module provides the Wi-Fi calling device with the inter-system service continuity mechanism driven by the service quality instead of the radio quality or the WLAN performance. V3 can be thus prevented. We estimate the voice quality based on the number of received voice packets per second on the device. Figure 18 plots the number of voice packets for a 140-second voice call. Since the Wi-Fi calling voice service uses the AMR (Adaptive Multi-Rate) audio codec, the packet rate varies with time. However, we observe that the packet rate is never smaller than 10 packets every two seconds. This rate can be thus used to detect whether the device is being under service degradation attacks. Once any suspicious attack is detected at the client or the server, the inter-system service continuity mechanism is triggered.

7.2 Implementation

The client of Wi-Fi Calling Guardian is an Android application written in Java and implemented on a Google Pixel

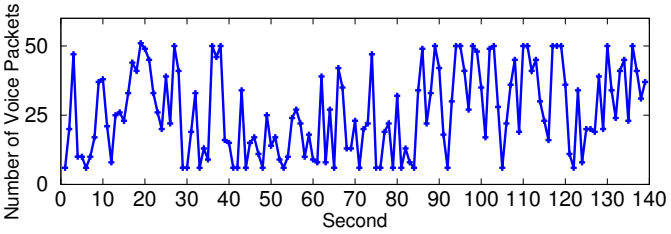
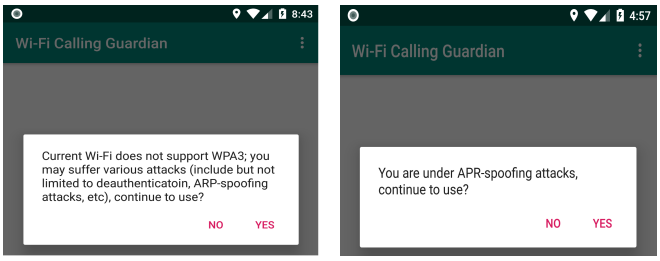


Fig. 18. The voice packets sent from the phone per second.



(a) WPA3 is not detected (b) Under an ARP spoofing attack

Fig. 19. The Wi-Fi security examiner detects the WPA3 usage and any ongoing ARP spoofing attack.

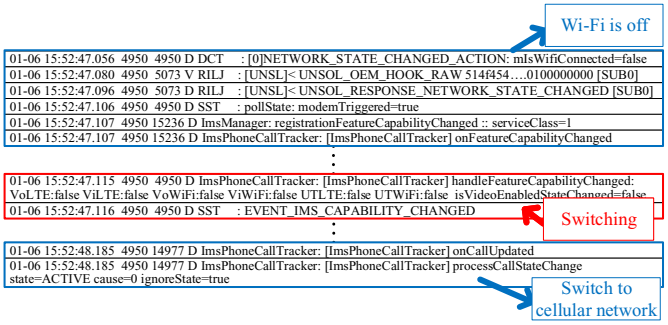


Fig. 20. A log from the Android logcat shows that a voice call over Wi-Fi calling is switched to the cellular-based voice based on the Wi-Fi disabling.

XL with a CPU, Qualcomm Snapdragon 821, and a 4GB RAM; the server is a network program written in C++ and implemented on a Dell precision tower 5810 with a CPU, E5-1603, and an 8GB RAM. We next elaborate on the implementation of each key component.

Wi-Fi security examiner: This module is implemented in the client with two detection mechanisms. First, the module uses an Android class of `WiFiManager` to obtain the Wi-Fi connection status that indicates whether the WPA3 is enabled. Second, the module uses a command ``arp -a'' to access the ARP table of the client device, and then detects the ARP spoofing attack by checking whether any two entries share the same MAC address.

VPN: We use OpenVPN to set up the VPN tunnel between the client and the server. On the client side, only Wi-Fi calling packets are forwarded through the VPN connection, whereas the other packets are directly routed to their destinations. Since the Android system does not allow the OpenVPN client to redirect the packets from a system application (i.e., the Wi-Fi calling application), we deploy the OpenVPN client on a software-based Wi-Fi router to which the client device connects. Through the VPN tunnel, all the uplink packets of Wi-Fi calling are delivered to the service quality monitor on the server, whereas all the downlink packets of Wi-Fi calling are forwarded to the singularity rectifier on the client.

Singularity rectifier: Data padding or packet fragmentation is performed on each Wi-Fi calling packet so that the packets can be encapsulated into 300-byte UDP datagrams. This module is implemented using the Type-length-value encoding scheme for the packets. Specifically, five types of the UDP payload are developed: (1) *signaling-packet*, which specifies the start and stop of the normalized data transmission; (2) *original-packet*, which contains a complete IPSec packet; (3) *fragment-packet*, which contains a complete IPSec header, a fragment of an IPSec packet, and the fragment's sequence number; (4) *padding-data*, which contains padding data; (5) *inter-system-switch-request*, which carries an inter-system switch request for the Wi-Fi calling service. After the IPSec packets are restored from the UDP datagrams, they are forwarded to the service quality monitor.

Service quality monitor: When the number of received small Wi-Fi calling packets is smaller than 10 during two seconds or a request of the inter-system switch is received, this module triggers the inter-system switch by disabling the device's Wi-Fi interface via an Android class of `WiFiManager`. Without the Wi-Fi access, the Wi-Fi calling device can be automatically switched back to the cellular network.

7.3 Evaluation

We next evaluate the performance of those three key components and present a small-scale user study.

Wi-Fi security examiner: We deploy a test Wi-Fi network which does not support the WPA3 protocol, and make the smartphone of Google Pixel XL connect with the Wi-Fi network. We further launch an ARP spoofing attack against all the devices from a computer in the Wi-Fi network. Figure 19 shows the evidence that the client of Wi-Fi Calling Guardian on the smartphone can successfully detect a lack of WPA3 and the ARP spoofing attack.

Singularity rectifier: We evaluate whether the singularity rectifier can defend against the THDoS and user privacy leakage attacks. The experiment is conducted as follows. First, we dial a Wi-Fi calling call from one device to another device with the client of Wi-Fi Calling Guardian, where the singularity rectifier is enabled. Second, we launch the annoying-incoming-call attack that discards the `180 Ringing` message and causes the caller device to get stuck in the dialing screen (see Section 5). Third, we use the WiCA module to infer the call statistics of this call from the callee's connected Wi-Fi network. Our experimental result shows that the singularity rectifier can well defend against those two attacks. Specifically, in the first attack, the `180 Ringing` message cannot be identified because no large IPSec packets (800-1360 bytes) are observed. In the second attack, $T_{RingingStart}$ and $T_{TalkingStart}$ are not identified due to a lack of the *C-Large* IPSec packets. Therefore, the ringing time and the call conversation time cannot be inferred.

Service quality monitor: We evaluate whether the service quality monitor can detect an attack of the service quality degradation and then initiate a inter-system switch of the

Wi-Fi calling service continuity. We launch a telephony denial-of-voice-service attack which discards 70% packets of a Wi-Fi calling call against a device after the call conversation starts. Our experimental result shows that the service quality monitor can detect the service quality degradation within 2 seconds after the attack is launched, and immediately trigger the inter-system switch, which is finished within 1 second as shown in Figure 20.

User study: To examine whether the VPN-based approach can significantly downgrade the voice quality of the Wi-Fi calling calls, we invite 10 students to participate in a user study experiment. In the experiment, we dial two Wi-Fi calling calls to each participant. One call is made with enabling Wi-Fi Calling Guardian, whereas the other is performed without it. The participants should report which one's voice quality is better or they are indistinguishable. Our experimental result shows that all participants cannot distinguish VPN-enabled Wi-Fi calling calls from original Wi-Fi calling calls (they think that both types of calls are the same in terms of voice quality), which means that Wi-Fi Calling Guardian does not downgrade the voice quality to a noticeable extent.

8 Related Work

Cellular Network Security: Cellular network security is getting more attention in recent years. Christian et al. [41] proposed Sonar to detect SS7 redirection attacks with audio-based distance bounding. Reaves et al. [42] introduced AuthentiCall to protect voice calls made over traditional telephone networks by leveraging now-common data connections available to call endpoints. Another study [43] analyzed nearly 400,000 text messages sent to public online SMS gateways over the course of 14 months and offered insights into the prevalence of SMS spam and behaviors. Jover [44] summarized the current state of affairs in the 5G protocol security and discussed the related areas that can be improved further. He et al. [45] presented a comprehensive survey of the attacks including RF jamming, signaling attacks, various SIP attacks, etc., in the LTE network. The other three works [46]–[48] study various attacks for SIP on different levels, discuss a potential attack based on SIP signaling, and classify existing SIP attacks and defenses, respectively. Compared with them, our work focuses on the security of the newly deployed Wi-Fi calling service security, which has not been fully explored yet.

VoIP and VoLTE Security: The security problem of the VoIP and VoLTE system has attracted lots of attentions. Two studies [49], [50] examine side-channel attacks on VoIP traffic. McGann et al. [51] analyzed the security threats and tools in the VoIP system. Several security issues (e.g., Toll Fraud) of VoIP applications were discussed in [52]. Li et al. [10] examined the security implications of VoLTE, which include several vulnerabilities (e.g., improper charing policies). Dacosta et al. [53] proposed the use of a modified version of OpenSER to improve authentication performance of distributed SIP proxies. This paper studies the Wi-Fi calling service from the perspectives of the standard, the implementation, and the operation, which are not covered by the prior arts.

Side-Channel Attacks Against Mobile Systems: The side-channel information leakage against mobile systems has been a popular research area in recent years. Current studies [49], [50] target the side-channel information leaked by mobile users' traffic, which is generated by some particular Internet services, and then seek to infer users' activities. The work [54] introduces the analysis on automatic fingerprinting of mobile applications for arbitrarily small samples of Internet traffic. Ali et al. [55] illustrated that each app leaves a fingerprint on its traffic behavior (e.g., transfer rates, packet exchanges, and data movement). Another work [56] demonstrates automatic fingerprinting and real-time identification of Android applications from their encrypted network traffic, which even could work when HTTPS/TLS is employed. Eskandari et al. [57] analyzed the personal data transfers in mobile apps and revealed that 51% of these apps did not provide any privacy policy. The paper [58] demonstrates discerning of mobile user location within commercial GPS resolution by leveraging the ability of mobile device magnetometers to detect externally generated signals in a permissionless attack. Reaves et al. [59] did the security analysis on the branchless banking applications. Different from them, we focus on the insecurity of the cellular Wi-Fi calling service, which is stipulated by the 3GPP and is going to be deployed globally on billions of mobile devices in the near future.

Wi-Fi Security: There are many novel studies related to Wi-Fi security. Liu et al. [60] used the fine-grained channel information to authenticate the user. Lee et al. [61] examined the limitations of the existing jamming schemes against channel hopping Wi-Fi devices in dense networks. Li et al. [62] inferred user demographic information by exploiting the meta-data of Wi-Fi traffic. Another study [63] proposes the system, the Wi-Fi Privacy Ticker, to improve participants' awareness of the circumstances in which their personal information is transmitted. Mikhail et al. [64] proposed an SBN model to effectively detect intrusions in the enterprise networks and the 802.11 wireless networks. Kolias et al. [65] categorized and evaluated popular attacks on the 802.11 networks, and applied different learning models to the collected dataset for the intrusion detection. Different from the prior art, our work investigates the insecurity of the Wi-Fi calling services, which have been deployed worldwide by cellular network operators, instead of new Wi-Fi vulnerabilities.

Wi-Fi Calling Security: Wi-Fi calling security is a new research area and has not been fully studied by the academic yet, since carriers just deployed their Wi-Fi calling services in recent years. Current researchers mainly focus on the security vulnerabilities on Wi-Fi calling devices. Specifically, Beekman et. al pointed out that T-Mobile Wi-Fi calling devices (e.g., Samsung S2) are vulnerable to invalid server certificates [66]. Chalakkal et. al studied SIM-related security issues on Wi-Fi calling devices [67]. However, our work examines the Wi-Fi calling security from two aspects: standards and operations.

9 Conclusion

The Wi-Fi calling service is thriving and being deployed worldwide. In this work, we conduct the first study on

the security implication of the operational Wi-Fi calling service over five operational networks, three in U.S. and two in Taiwan, using commercial Wi-Fi calling devices (e.g., Google Nexus 6P, Apple iPhone 8, Samsung Galaxy S8). We discover three security vulnerabilities which stem from design defects of the Wi-Fi calling standard (V1 and V3) and an operational slip of the Wi-Fi calling services (V2). By exploiting the vulnerabilities, adversaries are able to launch the telephony harassment or denial of voice service attack and infer the Wi-Fi calling user's privacy. In the attack of telephony harassment/DoS attack, the adversaries are able to shut the essential voice/text services down on the victims' smartphones while the security defenses deployed by Wi-Fi calling service providers and device manufacturers are suppressed. In the attack of user privacy leakage, adversaries can infer user identity, call statistics, device information, personalities, mood, malicious behaviors, etc.

The fundamental issue is that the conventional security defenses well examined in cellular network services are simply applied to the Wi-Fi calling service without considering its specific security threats. We thus develop a solution, called Wi-Fi Calling Guardian, which alleviates real-world damage by getting to the root of the vulnerabilities. The Wi-Fi calling service is still at its early rollout, so the lessons learned from the operational Wi-Fi calling service operators can help secure mobile ecosystem and facilitate the global deployment, as well as provide new design insights for upcoming 5G networks. We hope that our initial study will stimulate more research efforts on the Wi-Fi calling service from both academia and industry.

Acknowledgments

We greatly appreciate the anonymous reviewers' suggestions, which help to significantly improve this paper. We also thank Sihan Wang, Xinyu Lei, Wen Zhong, Jiawei Li, et al. for their participation in our experiments. This work is supported in part by the National Science Foundation under Grants No. CNS-1814551 and CNS-1815636. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors only and do not necessarily reflect those of the National Science Foundation.

References

[1] GSMA, "IR.51 IMS OVER WI-FI V5.0," May 2017, //www.gsma.com/newsroom/all-documents/ir-51-ims-wi-fi-v5-0/.

[2] 3GPP, "TS23.237:IP Multimedia Subsystem (IMS) Service Continuity; Stage 2," 2017.

[3] H. Telecoms, "Status of 4G/LTE and LTE-A networks globally," 2019, http://www.haddentelecoms.com/sites/default/files/2019-02/Status-of-LTE-networks-globally-02-2019.pdf.

[4] M. Intelligence, "VOICE OVER WIFI (VOWIFI) MARKET - GROWTH, TRENDS, AND FORECAST (2020 - 2025)," 2019, https://www.mordorintelligence.com/industry-reports/voice-over-wifi-vowifi-market.

[5] M. Garcia-Martin, "Input 3rd-Generation Partnership Project (3GPP) Release 5 Requirements on the Session Initiation Protocol (SIP)," 2005, https://tools.ietf.org/html/rfc4083.

[6] R. Jesske, D. Telekom, K. Drage, and C. Holmberg, "Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3GPP," 2014, https://tools.ietf.org/html/rfc7315.

[7] 3GPP, "TS33.401: 3GPP SAE; Security architecture," Sep. 2013.

[8] ——, "TS24.302:Access to the 3GPP Evolved Packet Core (EPC) via non-3GPP access networks," 2017.

[9] ——, "Ts23.402: Architecture enhancements for non-3gpp accesses;."

[10] C.-Y. Li, G.-H. Tu, C. Peng, Z. Yuan, Y. Li, S. Lu, and X. Wang, "Insecurity of voice solution volte in lte mobile networks," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 316–327.

[11] S. Bhattarai, S. Rook, L. Ge, S. Wei, W. Yu, and X. Fu, "On simulation studies of cyber attacks against lte networks," in *Computer Communication and Networks (ICCCN), 2014 23rd International Conference on*. IEEE, 2014, pp. 1–8.

[12] Y.-A. de Montjoye, J. Quoidbach, F. Robic, and A. S. Pentland, "Predicting personality using novel mobile phone-based metrics," in *International conference on social computing, behavioral-cultural modeling, and prediction*. Springer, 2013, pp. 48–55.

[13] S. Thomée, A. Härenstam, and M. Hagberg, "Mobile phone use and stress, sleep disturbances, and symptoms of depression among young adults-a prospective cohort study," *BMC public health*, vol. 11, no. 1, p. 66, 2011.

[14] V. Balasubramaniyan, M. Ahamad, and H. Park, "Callrank: Combating SPIT using call duration, social networks and global reputation," in *CEAS'07*, 2007.

[15] D. Sisalem, J. Kuthan, and S. Ehlert, "Denial of service attacks targeting a sip voip infrastructure: attack scenarios and prevention mechanisms," *IEEE Network*, vol. 20, no. 5, pp. 26–31, 2006.

[16] J. Tang, Y. Cheng, Y. Hao, and W. Song, "Sip flooding attack detection with a multi-dimensional sketch design," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 6, pp. 582–595, 2014.

[17] D. Geneiatakis and C. Lambrinoudakis, "An ontology description for sip security flaws," *Computer Communications*, vol. 30, no. 6, pp. 1367–1374, 2007.

[18] 3GPP, "TS23.401: GPRS Enhancements for E-UTRAN Access," 2011.

[19] J. Arkko, V. Lehtovirta, P. Eronen, R. G. Authentication, K. Agreement *et al.*, "Extensible authentication protocol method for 3rd generation authentication and key agreement (eap-aka)", rfc 4187," 2006.

[20] 3GPP, "TS33.402: Security aspects of non-3GPP accesses," Jun. 2018.

[21] V. Devarapalli and F. Dupont, "Mobile ipv6 operation with ikev2 and the revised ipsec architecture," RFC 4877, April, Tech. Rep., 2007.

[22] GSMA, "Wi-Fi Roaming Guidelines Version 12," September. 2017.

[23] CISION, "comScore Reports June 2017 U.S. Smartphone Subscriber Market Share," 2017, https://www.prnewswire.com/news-releases/comscore-reports-june-2017-us-smartphone-subscriber-market-share-300498296.html.

[24] W.-F. Alliance, "Hotspot 2.0 Specification," 2019. [Online]. Available: https://www.wi-fi.org/downloads-registered-guest/Hotspot_2.0_Specification_Package_v2.0.zip/29728

[25] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[26] UCLA, "Cellular Network Trace Collector: Spurring In-Phone Mobile Network Intelligence," 2017, http://www.mobileinsight.net/.

[27] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[28] P. Hu and D. Ramanan, "Finding tiny faces," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 1522–1530.

[29] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[30] L. Tran, X. Yin, and X. Liu, "Disentangled representation learning gan for pose-invariant face recognition," in *CVPR*, vol. 3, no. 6, 2017, p. 7.

[31] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 3. IEEE, 2006, pp. 850–855.

[32] M. Cristani, A. Pesarin, A. Vinciarelli, M. Crocco, and V. Murino, "Look at who's talking: Voice activity detection by automated gesture analysis," in *Constructing Ambient Intelligence*, R. Wichert, K. Van Laerhoven, and J. Gelissen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 72–80.

[33] "scikit-learn: Machine Learning in Python," http://scikit-learn.org/stable/.

[34] VLFeat, "VLFeat 0.9.21," 2018, http://www.vlfeat.org/.

[35] MSU, "MSU High Performance Computing." [Online]. Available: https://wiki.hpcc.msu.edu/

[36] IEEE, "Precision Time Protocol," 2008, https://en.wikipedia.org/wiki/Precision_Time_Protocol.

[37] S. CARLO, "Britain has more surveillance cameras per person than any country except china. that is a massive risk to our free society," https://time.com/5590343/uk-facial-recognition-cameras-china/, 2019.

[38] zhihu, "Do you know what level of domestic face recognition monitoring is achieved," https://zhuanlan.zhihu.com/p/39868461, 2019.

[39] G. Barber, "Some us cities are moving into real time facial surveillance," https://www.wired.com/story/some-us-cities-moving-real-time-facial-surveillance/, 2019.

[40] W.-F. Alliance, "Wpa3 specification version 1.0," 2019.

[41] P. Christian, A. Hadi, S. Nolen, B. Jasmine, T. Patrick, R. Bradley, and B. Kevin, "Sonar: Detecting ss7 redirection attacks with audio-based distance bounding," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 86–101.

[42] B. Reaves, L. Blue, H. Abdullah, L. Vargas, P. Traynor, and T. Shrimpton, "Authenticall: Efficient identity and content authentication for phone calls," in *26th USENIX Security Symposium (USENIX Security 17)*, 2017, pp. 575–592.

[43] B. Reaves, N. Scaife, D. Tian, L. Blue, P. Traynor, and K. R. Butler, "Sending out an sms: Characterizing the security of the sms ecosystem with public gateways," in *Security and Privacy (SP), 2016 IEEE Symposium on*. IEEE, 2016, pp. 339–356.

[44] R. P. Jover, "The current state of affairs in 5g security and the main remaining security challenges," *arXiv preprint arXiv:1904.08394*, 2019.

[45] L. He, Z. Yan, and M. Atiquzzaman, "Lte/lte-a network security data collection and analysis for security measurement: a survey," *IEEE Access*, vol. 6, pp. 4220–4242, 2018.

[46] D. Geneiatakis, T. Dagiuklas, G. Kambourakis, C. Lambrinoudakis, S. Gritzalis, S. Ehlert, and D. Sisalem, "Survey of security vulnerabilities in session initiation protocol," *IEEE Communications Surveys and Tutorials*, vol. 8, no. 1-4, pp. 68–81, 2006. [Online]. Available: https://doi.org/10.1109/COMST.2006.253270

[47] D. Geneiatakis, G. Kambourakis, T. Dagiuklas, C. Lambrinoudakis, and S. Gritzalis, "Sip security mechanisms: A state-of-the-art review," in *In Proc. 5th International Network Conference (INC*. ACM, 2005, pp. 147–155.

[48] U. U. Rehman and A. G. Abbasi, "Security analysis of voip architecture for identifying sip vulnerabilities," in *Emerging Technologies (ICET), 2014 International Conference on*. IEEE, 2014, pp. 87–93.

[49] A. Compagno, M. Conti, D. Lain, and G. Tsudik, "Don't skype & type!: Acoustic eavesdropping in voice-over-ip," in *AsiaCCS*. ACM, 2017, pp. 703–715.

[50] J. Fang, Y. Zhu, and Y. Guan, "Voice pattern hiding for voip communications," in *Computer Communication and Networks (ICCCN), 2016 25th International Conference on*. IEEE, 2016, pp. 1–9.

[51] S. McGann and D. C. Sicker, "An analysis of security threats and tools in sip-based voip systems," in *Second VoIP security workshop*, 2005.

[52] P. C. Hung and M. V. Martin, "Security issues in voip applications," in *Electrical and Computer Engineering, 2006. CCECE'06. Canadian Conference on*. IEEE, 2006, pp. 2361–2364.

[53] I. Dacosta, V. Balasubramaniyan, M. Ahamad, and P. Traynor, "Improving authentication performance of distributed sip proxies," in *Proceedings of the 3rd International Conference on Principles, Systems and Applications of IP Telecommunications*. ACM, 2009, p. 1.

[54] E. Bocchi, L. Grimaudo, M. Mellia, E. Baralis, S. Saha, S. Miskovic, G. Modelo-Howard, and S.-J. Lee, "Magma network behavior classifier for malware traffic," *Computer Networks*, vol. 109, pp. 142–156, 2016.

[55] A. I. Ali-Gombe, B. Saltaformaggio, D. Xu, G. G. Richard III *et al.*, "Toward a more dependable hybrid analysis of android malware using aspect-oriented programming," *computers & security*, vol. 73, pp. 235–248, 2018.

[56] V. F. Taylor, R. Spolaor, M. Conti, and I. Martinovic, "Robust smartphone app identification via encrypted network traffic analysis," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 63–78, 2018.

[57] M. Eskandari, B. Kessler, M. Ahmad, A. S. de Oliveira, and B. Crispo, "Analyzing remote server locations for personal data transfers in mobile apps," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 1, pp. 118–131, 2017.

[58] K. Block and G. Noubir, "My magnetometer is telling you where i've been?: A mobile device permissionless location attack," in *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. ACM, 2018, pp. 260–270.

[59] B. Reaves, J. Bowers, N. Scaife, A. Bates, A. Bhartiya, P. Traynor, and K. R. Butler, "Mo (bile) money, mo (bile) problems: analysis of branchless banking applications," *ACM Transactions on Privacy and Security (TOPS)*, vol. 20, no. 3, p. 11, 2017.

[60] H. Liu, Y. Wang, J. Liu, J. Yang, Y. Chen, and V. Poor, "Authenticating users through fine-grained channel information," *IEEE Transactions on Mobile Computing*, no. 1, pp. 1–1, 2018.

[61] I.-G. Lee, H. Choi, Y. Kim, S. Shin, and M. Kim, "Run away if you can: Persistent jamming attacks against channel hopping wi-fi devices in dense networks," in *International Workshop on Recent Advances in Intrusion Detection*. Springer, 2014, pp. 362–383.

[62] H. Li, Z. Xu, H. Zhu, D. Ma, S. Li, and K. Xing, "Demographics inference through wi-fi network traffic analysis," in *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*. IEEE, 2016, pp. 1–9.

[63] S. Consolvo, J. Jung, B. Greenstein, P. Powledge, G. Maganis, and D. Avrahami, "The wi-fi privacy ticker: improving awareness & control of personal information exposure on wi-fi," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*. ACM, 2010, pp. 321–330.

[64] J. W. Mikhail, J. M. Fossaceca, and R. Iammartino, "A semi-boosted nested model with sensitivity-based weighted binarization for multi-domain network intrusion detection," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 3, p. 28, 2019.

[65] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 184–208, 2015.

[66] J. Beekman and C. Thompson, "Man-in-the-middle attack on t-mobile wi-fi calling," *Electrical Engineering and Computer Sciences University of California at Berkeley, https://www2.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-18.pdf*, 2013.

[67] S. Chalakkal, H. Schmidt, and S. Park, "Practical attacks on volte and vowifi," *ERNW Enno Rey Netzwerke, Tech. Rep*, 2017.